SWTPC 6800 DISASSEMBLER

BY MICHAEL J. DENNIS

The DESEMBLER program can be used to "de-assemble" a loaded machine code program. This may be desirable to aid debugging or to assist in understanding, correcting or modifying a machine code supplied program. The DESEMBLER lists the memory address of each byte of the instruction, the equivalent ASCII character of each byte, the assembly language mnumonic of the machine code instruction along with addressing symbols and the operand. Some of the features of this program are:

The DESEMBLER can be run from any location in memory.

Program has a built in "move" routine which will move the DESEMBLER to the desired memory location.

Program has a "byte search" option which will find every occurance of a specified byte within a memory area and print it out with its address, the preceeding byte, and the following two bytes.

The DESEMBLER has an optional "intersection mark" feature which marks in the desembled listing the program locations to which the program branches or jumps to. These marks indicate whether it was a direct branch or a subrountine type branch. This option will also cause a symbols table listing of the intersection addresses to be printed after the regular desembler listing.

Output may be directed to any port via a control, asynchronous (ACIA), or parallel (PIA) interface.

No memory locations need be initialized prior to running the DESEMBLER other than the program counter (A048-A049).

Hardware configuration and software support:

This program is designed to run in any 2.5K of continous memory on a SWTPC 6800 computer system. Program requires MIKBUG* software and much of the MIKBUG* RAM memory (A000-A07F). The DESEMBLER communicates with the control terminal at port #1 and asynchrounos serial output devices such as a TELETYPE* or parallel output devices such as a PR-40 printer are supported as options at any other port. The DESEMBLER as supplied initially loads into memory locations 0100-0AFF.

*MIKBUG is a registered trademark of Motorola Inc.
*TELETYPE is a registered trademark of Teletype Inc.

Operation of the DESEMBLER:

Upon loading A048 and A049 with the starting address (0100) and typing the MIKBUG* command "G" to start, the DESEMBLER will identify itself on the control terminal and ask for the option wanted as shown.

```
SWTPC DESEMBLER V.4

OPTION (D,S,M,E):
```

The user should then type a D, S, M, or E where:

D indicates a desembled listing is desired. (disassembly)
S for byte search routine.
M for moving DESEMBLER program to new memory location.
E for exiting back to MIKBUG*.

If "E" is entered, the program immediately returns to MIKBUG*. To restart DESEMBLER type MIKBUG* command "G".

If "M" is typed the program will ask:

```
NEW BEGINNING ADDRESS:
```

The user should key in the desired 4 digit hex address that the program should be moved to. The program will restart upon completion at the new location. When a move is made, be sure that sufficient memory exists for the program and for the creation of the intersection data file (symbol table) if intersection marks are to be printed. Intersections point to places in the program to where a branch or jump instruction terminates.

If the "D" option is chosen, the DESEMBLER will ask:

```
MARK INTERSECTIONS? (Y/N)
```

If the user wants the program to indicate the address locations which the desembled program branches or jumps to, he should type a "Y" for yes, otherwise "N" should be typed for no. In the listing, locations that are referred to by a BSR or JSR instruction will be flagged by a >. Locations referred to by a branch instruction are flagged by a -. The DESEMBLER will then ask:

```
INPUT FILE ADDRESS:
```

The user should key in the 4 digit hex beginning address of the area of memory to be desembled. The program will then type "TO" and the user should key in the ending address of this memory area. The DESEMBLER will than ask:

```
PORT & DEVICE (0-7; C,S,P):
```

The user should respond with the port number and the output device type where "C" is the control interface, "S" is a serial (ACIA) interfaced device, and "P" indicates a parallel interfaced printer. Note that if "C" is chosen, output will be to port #1 regardless of the entered port number.

The DESEMBLER will then type:

```
HEADER:
```

The user should respond with the title to be printed on the desembled listing, up to 44 characters in length, terminated by a carriage return.

If "S" was the option chosen, indicating a "byte search", the program will ask for the input file address and the port and device as it would for the "D" option. The program will then ask for the "BYTE DESIRED:". The user should enter the 2 digit hex number. The DESEMBLER will then output to the selected output device the address of each occurance, the preceeding byte, the byte of interest (corresponding to

the address), and the following two bytes. To exit from this routine a reset of carriage return must be hit. To restart program type the MIKBUG* command "G".

Input notes:

If an illegal option or output device is entered, the DESEMBLER will re-ask the question. If an error occurs in the input of a number, however, program control returns to MIKBUG*. The command "G" will restart the DESEMBLER. If too many characters are entered in the header title, the carriage return will be assumed as the 44th character and output will begin.

Output notes:

The "desemble" option of the DESEMBLER will produce a pagenated output including cut marks, the header title, and the page number. The page will list 56 (38 hex) lines which include the memory address of the instruction, the memory contents and ASCII equivalent of each byte in the instruction, the instruction mneumonic, addressing symbols and operand. The addressing symbols and operand are in the following format:

blank	for inherent addressing
# 3E or # AØ3E	for immediate addressing
AØ3E or ØØ3E	for extended or direct addressing
X, 3E	for indexed addressing
Ø23E	for relative addressing

Note that all numbers are hexadecimal. Also note that relative addressing gives the absolute address of the branch rather than the relative address contained in the next memory location.

Next to the memory contents in the DESEMBLER output is printed the ASCII equivalent of the byte or a space. A space is printed for all control characters or non-capital ASCII characters. Whether or not the equivalent ASCII character is printed out for other values depends on the parity of the memory contents (the most significant bit) and on the instruction at the relative address of Ø286 (hex) in the DESEMBLER program. If the starting address of the DESEMBLER + 286 (hex) contains

- 2Ø then no ASCII characters will be printed,
- 2B then ASCII characters with parity=Ø will be printed (default),
- 2A then ASCII characters with pari-y=1 will be printed,
- 25 then all ASCII characters will be printed regardless of parity.

The number of lines printed per page is controlled by the number in the relative address 246 of the DESEMBLER program (normally 38 hex). (Ø286 relative is location Ø386 and Ø246 relative is Ø346 if the DESEMBLER is origined at Ø1ØØ as supplied. When a move is done the locations will naturally change.)

Below is a sample run of the DESEMBLER with some comments on the output.

NOTE Location ØA14 in the object dump should be 42 and location ØA15 should be C9.

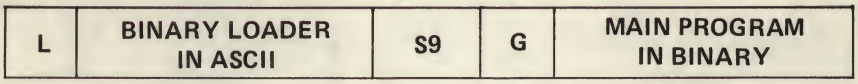
```
*G SWTP 6800 DESEMBLER V.4

OPTION (D,S,M,E): D
MARK INTERSECTIONS? (Y/N) Y

INPUT FILE ADDRESS: 0100 TO 0175
PORT & DEVICE (0-7; C,S,P): 7P

HEADER: DESEMBLER VER. 4.0

====


DESEMBLER VER  4 0          PAGE 01


0100 34        4              DES
0101 34        4              DES
0102 20 6B                    BRA    : 016F
0104 FE A048     H          - LDX      A048
0107 FF A014                  STX      A014
010A 86 71                    LDA A #  71
010C 08                     - INX
010D 4A        J              DEC A
010E 26 FC     &              BNE    : 010C
0110 FF A016                  STX      A016
0113 CE A054     T            LDX    # A054
0116 FF A04E     N          - STX      A04E
0119 FE A016                  LDX      A016
011C A6 00                    LDA A X.   00
011E 08                       INX
011F FF A016                  STX      A016
0122 FE A04E     N            LDX      A04E
0125 A7 00                    STA A X    00
0127 08                       INX
0128 81 3B       ;            CMP A #  3B
012A 26 EA     &              BNE    : 0116
012C 86 06                    LDA A #  06
012E 8D 67                    BSR    : 0197
0130 BD E047     G            JSR      E047
0133 FF A04A     J            STX      A04A
0136 FF A04C     L            STX      A04C
0139 FF A048     H            STX      A048
013C 5F        +              CLR B
013D B6 A04A     J            LDA A    A04A
0140 B1 A014                  CMP A    A014
0143 2B 0A     +              BMI    : 014F
0145 2E 13                    BGT    : 015A
0147 B6 A04B     K            LDA A    A04B
014A B1 A015                  CMP A    A015
014D 2A 0B     *              BPL    : 015A
014F 5C        \            - INC B
0150 86 0A                    LDA A #  0A
0152 BB A04C     L            ADD A    A04C
0155 B7 A04C     L            STA A    A04C
0158 20 10                    BRA    : 016A
```

```
S1130100343420 6BFEA048FFA0148671084A26FCF4
S113011 0FFA016CEA054FFA04EFEA016A60008FF16
S113012 0A016FEA04EA70008813B26EA86068D672E
S113013 0BDE047FFA04AFFA04CFFA0485FB6A04A1D
S113014 0B1A0142B0A2E13B6A04BB1A0152A0B5C38
S113015 0860ABBA04CB7A04C2010860ABBA014B7DB
S113016 0A014860ABBA04AB7A04A7EA054209520BA
S113017 032FEA014A6008D14FFA014FEA04AA7000E
S113018 0BCA04C270E8D05FFA04A20E55D26020980
S113019 03908398EA0423B206020D208080804A5A
S11301A 026F939FEA04886E48DF1FFA018860D8D4E
S11301B 0EAFFA01A86178DE3FFA01C8629 8DDCFFB9
S11301C 0A01E860D8DD5FFA0208641 8DCEFFA022D6
S11301D 086018DC78DC5FFA024CE8004FFA028868C
S11301E 001B7A02A86028D11BDE1ACB7A052814DA2
S11301F 027A78145264 87EE0D0204186 0A8D3DFE12
S1130200A0208 60C084A26FCBDE07EBDE05516FE03
S1130210A0140908A600112619FFA04ACEA04A86F8
S113022 0F08D19FEA04A098D108D0E860A8D0DFEE3
S1130230A04ABCA01626DC20C28D004F2061814458
S1130240270681532693201086078D53BDE1AC5FAA
S1130250815926015CF7A02C86038D43BDE047FF3E
S1130260A0148604 8D39BDE047FFA0168605 8D2FA6
S1130270BDE0AA81082AF5484816BDE1AC814327B0
S113028 02E8153271C815026E38D0D86FFB7A02AAB
S1130290A700863FA7012017F7A029FEA028392030
S11302A0438DF54FB7A02A8613A7008611A700B681
S11302B0A052815327817FA050FEA020BDE07ECEB6
S11302C0A054BDE1AC810D2708A70008CA07F26AF
S11302D0F18604A700FEA01CAD00FEA014FFA04AF6
S11302E08D462018 2065F6A0515C2004860A8D5B9B
S11302F05A26F98D33FEA01A6E008D088D4F8D7924
S1130300 8D1720F67AA05126028D1DFEA04A8D007D
S1130310 8D0009BCA01627CE39B6A02DFEA04A4AEE
S113032 027F608FFA04A20F786088D1FCEA0548D1B
S1130330 69860B8D16860D8D12CEA050A6008B01FA
S1130340 19A7008D5E8638A700860920438 60A8D8A
S1130350 3FCEA04A86F08D38FEA01EAD00F6A02D3B
S1130360 FEA04A8D168D478D128D108D3FF6A02D5F
S1130370 FEA04A8D0B8D092007202 65A2B2E202300
S1130380 5A2B2BA600082B2648482822644202016
S1130390 4D2B6C276C206E8D1708A600810426F760
S11303A 039207AA6008D48A600082 0478D008620B3
S11303B 0FFA04C847FFEA0287DA02A2B172707BD11
S11303C 0E1D1FEA04C3937E600575724FAA7013390
S11303D 0FEA04C3937A700C637E701C63FE7016DD9
S11303E 0012AFCE60033FEA04C39A6000820C144D3
S11303F 0444444840F8B30813923B58B0720B18D5D
S1130400 A28DA08614FFA04EFEA02037E60008C1EE
S1130410 0426F94A26F68D8233FEA04E39860B8DCA
S1130420 E4FEA018AD0086148DDBFEA02FC6068D59
S1130430 B98DB78DB58DCC8DB18DC8B6A02E2601E2
S1130440 394A2609860C8DBDFEA04A20B24A2609E7
S1130450 860E8DB1FEA04A20A84A2622860F8DA5BD
S1130460 FEA04AA60008FFA052CEA0524D2A026A5E
S1130470 005FAB01E900E700A7012083208 74A263B
S1130480 04861020CD8611F6A02D5A5A27C420B612
S1130490 FEA04AFFA03108FFA04A7DA02C26013906
S11304A 08D54BCA026260 4860C20D18614C680E474
S11304B 00027024A4A8DF28614C640E40027014A66
S11304C 020E720B67DA02C2753FEA01409FFA04AE4
S11304D 0FEA04A08FFA04ABCA01626037EE0D0FF77
S11304E 0A0318D12BCA02627E78DC0CEA04A8DD2A4
S11304F 0860A8DB520DAB6A031843FB7A031FEA0BC
S1130500 2409090808BCA026260139A600843FB1A5
S1130510 A03126EFA601B1A03226E8397EE0D00052
S1130520 7DA02C260139FEA024FFA0266F006F01B8
S1130530 FEA014FFA04AA600817E272981BD272C96
S1130540 818D273784F0812027388D78FEA04AB624
S1130550 A02D08BCA01627084A26F7FFA04A20D6DB
S1130560 3920E720918680B7A02B20058640B7A0CC
S1130570 2BFEA04AEE01FFA03120248640B7A02B19
S1130580 20058680B7A02B08A60008FFA031CEA0C6
S1130590 314D2A026A005FAB01E900E700A7018D33
S11305A 0C2BCA026260B6F02086F0208FFA026091Z
S11305B 009B6A031BAA02BAA00A700B6A032A701A1
S11305C 0209F0000FEA04AE600FEA0225C08080866
S11305D 0085A26F9FFA02F8D138D111727044444C0
S11305E 0444C4CB7A02DC407F7A02E398D00A600AB
S11305F 048590839000000000D0A0A48454144459D
S1130600 523A20040D0A4259544520444553495254
S1130610 45443A2004101653575450203638303080
S1130620 20444553454D424C4552202056 2E340D0E
S1130630 0A0A0A4F5054494F4E2028442C532C4D3B
S1130640 2C45293A20040D0A0A494E505554204697
S1130650 494C4520414444524553533A20042054C4
S1130660 4F20040D0A504F5254202620444556492 9
S1130670 43452028302D373B20432C532C50293A16
S1130680 20040D0A4E4557204245474 94E4E494ED7
S1130690 472041444452455353 3A20040D0A4D41E6
S11306A 0524B20494E5445525345435449 4F4E539F
S11306B 03F2028592F4E2920040D0A0A0A0A2D2DFD
S11306C 02D2D0D0A0A0A0A040D0A0A040D0A000453
S11306D 020202020202020200420200450414745B1
S11306E 0200458 2C2020043E200420203030042 3F1
S11306F 020042D043E0420040000000000000003B
S1130700 3F20202 04E4F50203F2020203F202020FB
S1130710 3F20202 03F2020205441502054504120 8D
S1130720 494E582044455820434C562053455620A2
S1130730 434C432053454320434C492053454920CF
S1130740 53424120434241203F2020203F2020208B
S1130750 3F20202 03F2020205441422054424120 69
S1130760 3F20202 044414120 3F202020414241207D
S1130770 3F20202 03F2020203F2020203F2020203F202020F9
S1130780 4252C1A03F2020204248C9A0424CD3A0DD
S1130790 4243C3A04243D3A0424EC5A04245D1A088
S11307A 04256C3A04256D3A04250CCA0424DC9A049
S11307B 04247C5A0424CD4A04247D4A0424CC5A055
S11307C 0545358 20494E532050554C4150554C4297
S11307D 0444553205458532050534841505348 42A1
S11307E 03F2020205254532 03F2020205254 9209F
S11307F 03F2020203F2020205741492053574920A3
S1130800 4E4547413F2020203F202020434F4D416B
S1130810 4C5352413F202020524F524141535241A8
S1130820 41534C41524F4C41444543413F202020C9
S1130830 494E434154535441 3F202020434C52419C
S1130840 4E4547423F2020203F202020434F4D4229
S1130850 4C5352423F202020524F52424153524265
S1130860 41534C42524F4C42444543423F20202086
```

```
S1130870494E4342545354423F2020204340524259
S11308804E45C7203F2020203F2020204340CD202D
S11308904C53D2203F202020524FD2204153D2200B
S11308A04153CC20524FCC204445C3203F2020202C
S11308B0494EC3205453D4204A4DD020434CD22017
S11308C0CE4547A03F2020203F2020204C34F4DA0ED
S11308D0CC5352A03F2020204D24F52A0C15352A04B
S11308E0C1534CA0D24F4CA0C44543A03F2020206C
S11308F0C94E43A0D45354A0CA4D50A0C34C52A0D7
S1130900D53D542C143CD50C153C243C13F2020200F
S113091041CE44C142C954C14CC441C13F2020200EE
S113092045CF52C141C443C14FD241C141C444C166
S1130930C3D058A042532D2A0CCC453A03F2020200FF
S113094053D5424143CD504153C243413F2020201F
S113095041CE444142C954414CC4414153D4414124
S113096045CF524141C443414FD2414141C4444126
S11309704BD058203F2020204CC4532053D453202C
S1130980585355C241434DD0415342C3413F2020200DF
S1130990414EC4414249D4414C44C1415354C141E4
S11309A0454FD2414144C3414F52C1414144C441E6
S11309B04350D8204A53D2204C44D3205354D320FC
S11309C0D35542C1C34D50C1D34243C13F2020201F
S11309D0C14E44C1C24954C1CC4441C1D35441C1A4
S11309E0C54F52C1C14443C1CF5241C1C14444C1A6
S11309F0C35058A0CA5352A0CC4453A0D35453A0BC
S1130A0053D5420243CD50C253C243C23F2020200DB
S1130A1041CE44C2C24254C24CC441C23F2020200F1
S1130A2045CF52C241C443C24FD241C241C444C261
S1130A303F2020203F2020200CCC458A03F2020204D
S1130A4053D5424243CD504253C243423F2020201B
S1130A5041CE444242C954424CC4414253D441421F
S1130A6045CF524241C443424FD2414241C4444221
S1130A703F2020203F2020204CC4582053D458200D
S1130A805355C242434DD0425342C3423F2020200DB
S1130A90414EC4424249D4424C44C1425354C142DF
S1130AA0454FD2424144C3424F52C1424144C442E1
S1130AB03F2020203F2020204C44D8205354D820CD
S1130AC0D35542C2C34D50C2D34243C23F2020201B
S1130AD0C14E44C2C24954C2CC4441C2D35441C29F
S1130AE0C54F52C2C14443C2CF5241C2C14444C2A1
S1130AF03F2020203F2020200CC4458A0D35458A08D
```

NOTICE TO USERS OF SWTPC PAPER AND CASSETTE TAPES

In order to help reduce the time necessary to load programs through either a paper tape reader or an SWTPC AC-30 cassette interface, the longer tapes supplied from SWTPC will be furnished in a binary format instead of the conventional ASCII. At the beginning of each tape is a binary loader program that will load into the computer using the regular ASCII format. The program then executes itself and loads the main program in binary. Using this method, tapes will load in approximately 1/3 normal time. When using an SWTPC AC-30, **lock** the reader in the ON position and type L. For paper tapes readers, such as on an ASR-33 Teletype,® when the load stops after the binary loader has been loaded into the computer simply type G. This will execute the binary loader and the remainder of the tape will load into memory. Several "garbage" characters may be printed immediately after the binary loader loads in—this is normal. On cassette tapes, one side will be in conventional ASCII (side with long leader) and one side will be in binary. The tapes are formatted as follows:

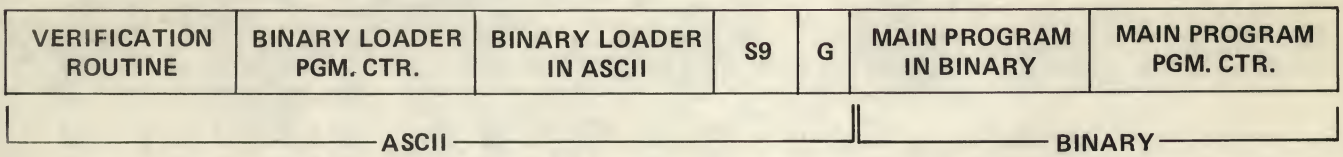


As the tape loads, you will see one of the following displays on your terminal: (either is OK)

*L	*L
*G	*??
* (register dump)	* (register dump)

Some tapes may have an additional feature which will verify that the tape loaded correctly into memory. If, after loading the tape, you find that the program counter is not automatically set to the correct value then you probably have a verifying tape. If this is the case simply typing a G will automatically check the validity of the program and execute it. If the message LOAD ERROR is displayed then the tape did not load correctly into memory. The most common cause of this is a memory problem—there can be problems that MEMCON and ROBIT will not find.

The format for a self-verifying tape is as follows:



As before, one side of a cassette tape will be in binary and the other side in ASCII.

If you are unable to load a tape please check the following:

1.) Be sure the reader is locked on to load a binary cassette tape.
2.) Try different volume and tone control settings.
3.) Clean your tape heads with alcohol and a cotton swab.
4.) Re-check all memory if a LOAD ERROR is displayed.